

Knowledge-based systems for the Configure Price Quote (CPQ) process – A case study in the IT solution business

Michelle Jordan, Leipzig University of Telecommunications (HfTL), Germany,
michellejordan@gmx.de

Gunnar Auth, Meissen University of Applied Sciences (HSF), Germany,
gunnar.auth@hsf.sachsen.de

Oliver Jokisch, Leipzig University of Telecommunications (HfTL), Germany,
jokisch@hft-leipzig.de

Jens-Uwe Kühn, T-Systems International GmbH, Germany,
jens-uwe.kuehl@t-systems.com

Abstract

Software systems for the Configure Price Quote (CPQ) process of complex product portfolios have emerged in the sales function of companies recently. A flexible quote of complex products, in particular for a Business-to-Business (B2B) customer requires a wide variability of product features and configurations, along with the ability to deliver competitive quotes in short time. The CPQ system aims to reduce the process time, to increase the process quality by integrating information and data stored in several enterprise systems with codified explicit and implicit knowledge from individuals. As in most of the knowledge management systems, the openness of the knowledge holders to share and codify their individual knowledge is a critical success factor. In this case study, we look at the CPQ system implementation of a multinational Information Technology (IT) solution provider from a process perspective and with regard to both the technical and organizational challenges in a holistic approach. The article starts with an introduction to CPQ systems based on works from the Knowledge Management (KM) domain. After outlining our research methodology, we present the case together with a generalization of the CPQ implementation process. Our findings from the investigated scenario indicate positive influence of 1) the internal promotion of CPQ systems as technology innovation for motivating expert knowledge holders to collaborate; 2) an active preparation of the organizational environment for the upcoming changes; and 3) a hybrid agile implementation process.

Keywords: Product configurator, sales configurator, CPQ system, hybrid agile, implementation process.

Introduction

In recent years, Commercial Off-The-Shelf (COTS) software products for supporting the Configure Price Quote (CPQ) process of complex product portfolios have emerged in the sales function of companies. Nowadays known as CPQ systems, these COTS products can be considered

as a further development of the first knowledge-based configuration systems from the end of the 1970s (Zanker & Tiihonen, 2008; Hotz et al., 2014). The dissemination of product configurators, originally started in the manufacturing industry, is closely connected to production strategies based on mass customization principles (Sorri et al., 2017). Offering customers tailored products requires a wide variability of product features and configurations. Offering tailored products for competitive prices additionally requires a flexible mass production system that enables economies of scale. Configuration systems are used for defining and planning valid product configurations from numerous possible variations and allow for rapid manufacturing of highly customized products (Helo, 2006).

Before production planners can think about how to build a customized product, this product has to be defined and calculated by a sales person or team in order to file a tender. To win the customer's order the tender must meet the customer requirements and offer a competitive quote. According to its main activities, this process is called "configure price quote". In the B2B domain, the CPQ process typically has to be performed by suppliers in response to a public invitation to tender or a request for quotation within a strictly limited bid period. It requires expert knowledge, information on product parts and features as well as production and material costs. In addition, product-related services might be included. Although the retrieval of the necessary information from various resources can already be difficult and time-consuming, the real bottleneck is often the individual (implicit and explicit) knowledge on configuration rules and possibilities within the minds of product and sales specialists. Dedicated CPQ software systems address these issues and aim to reduce the process lead-time as well as to increase the process quality by integrating information and data stored in enterprise systems with codified explicit knowledge from individuals.

Baltes et al. (2011) pointed out that the implementation of a CPQ solution means not only a technical but also an organizational challenge since it comes along with a radical change of existing sales and quotation processes. Similar to the implementation of general knowledge management systems, the openness of the knowledge holders to share and codify their individual knowledge is a critical success factor.

With this background, the focus of our article lies in the implementation of CPQ software systems from a process view. Our research goal is to understand how the process of implementing a CPQ solution should be designed to address both the technical and the organizational challenges in a holistic approach. Hence, we framed our research question as follows:

How should a strategy and respective process be designed in order to support the effective and efficient implementation of a knowledge-based system for the CPQ process based on COTS software?

Firstly, an overview is provided of the nature of CPQ systems and the relevant background literature. Then research methodology used is outlined, followed by the presentation of the case investigated. Finally, the findings of our research are presented with the discussion and conclusions.

CPQ Systems as an Emerging Type of Business Application Systems

The global phenomenon of digital transformation is pushing companies in many industries to continuously adapt in order to improve customer orientation. To withstand market pressure, it is essential to execute internal processes for preparing quotes as effectively and efficiently as possible. In particular, complex offer projects require rapid creation of a solution as well as meeting the customer's functional and quality requirements. Eventually, the customer will usually accept the offer with the best cost benefit ratio, while the supplier needs to end up with a sufficient margin. Therefore, an accurate cost calculation is essential for a successful offer.

The integration and optimization of the, formerly only loosely coupled, sales processes configure, price, and quote from the customer's point of view is the primary goal of CPQ application systems. While CPQ systems can eventually be designed for use by customers and thus implement the concept of self-service, our case investigates the specific issues of enterprise CPQ systems used by internal sales staff. As Business Application Systems (BAS) CPQ systems focus on the correspondent process for complex product and service portfolios, as in our case for offering tailored IT solutions to business customers, which combine hardware, software and network components as well as consulting and engineering services.

According to Ferstl and Sinz (2012), the purpose of a BAS is to automate or semi-automate a business process, parts of it or even several processes. Common types of BAS available as COTS products include Enterprise Resource Planning (ERP) (to automate processes), Customer Relationship Management (CRM) or Supply Chain Management (SCM). Besides their functional process orientation, CPQ systems can also be characterized from a technological viewpoint by their derivation from product configurators and recommender systems. In this respect, the development and growing popularity of CPQ systems is also driven by advances in Artificial Intelligence (AI) and Machine Learning (ML) (Kling, 2019; Sorri et al., 2017). Given these prerequisites, CPQ systems can be defined as a special type of BAS for automating the sales process of custom-tailored products, services or product-service bundles through integrating the main steps of configuration, pricing and creating the respective tender documents on the functional, process, data, and knowledge level. CPQ systems evolved from earlier product and sales configurators, but cover an extended sales-oriented scope and utilize recent technology innovations in AI/ML for configuration and recommendation.

A brief overview of current CPQ products in the market is given by Scott (2018). The addressed functionalities include the upload of product catalogs, and the setting of a "few rules" to standardize sales. Afterwards the sales team can access the catalog and quickly pull out a quote and deliver it to the customer. In standardized business segments, such quotes include bulk ordering, discounting, and time-dependent pricing structures, which reduce the calculation effort and facilitates accurate quotes for the customer. The commercial CPQ solutions mentioned such as CallidusCloud CPQ (now SAP), Oracle CPQ Cloud, Salesforce CPQ, Quotewerks, Endeavor CPQ, EOS CPQ are provided standalone, built into advanced CRM and sales tools, or as a feature in complex ERP suites. Based on more advanced methods of analytics, for example, by learning algorithms the functionality of CPQ systems is extended also for non-standard deal management.

Some of the key metrics include quote accuracy rate, quote turnaround time, standard to nonstandard deal ratio, quote conversion rate, and self-service quote ratio (Gill & Mathur, 2019).

The academic literature on knowledge-based configuration can be subdivided in contributions before and after the occurrence of the term CPQ as a name for an integrated process combining configuration, pricing and quotation activities as part of a sales process in business. The literature of the pre-CPQ era mainly concentrates on basic questions of how to design and to develop software systems for product configuration, so-called product configurators. Topics include, among others, knowledge representation and reasoning (Bowen, 1986; Franke, 1998), configuration models and ontologies (Asikainen et al., 2002; Krebs, 2006), problem-solving methods (Hadzic et al., 2004; Helo, 2006; Kramer, 1991), and service configuration (Heiskala et al., 2005; Tiihonen et al., 2006). Only a few authors have looked at configuration from a non-technical perspective, considering organizational aspects like business process improvement (; Baltes et al., 2011; Hvam et al., 2006). Hvam et al. (2006) were also among the first to propose the application of product configurators for the quotation process. Later Abbasi et al. (2013) described ‘the anatomy of a sales configurator’. Around that time the term CPQ also started to become popular in business practice (Ostrow, 2014), but was not used in academic literature yet (Hotz et al., 2014; Sorri et al., 2017). Until today there is very limited literature researching the evolution of sales configurators to CPQ solutions, although it has attracted quite a lot of attention in business practice and by analysts (Bruno, 2017; Klock & Lewis, 2019). Sorri et al. (2017) gave a general introduction to state-of-the-art CPQ system architectures and functionalities along with main characteristics of selected CPQ products. In this paper the authors intent to respond to this gap in academic literature.

Research Methodology

Since we were unable to find published method or valid approach for a CPQ implementation, we decided on a qualitative empirical research approach using a single case study as the primary research method. Our case explores the CPQ system implementation of a multinational IT solution provider, where we could analyze the period from start to end of the initial implementation project.

According to Yin (2013), case study research is appropriate to answer ‘how’ and ‘why’ questions related to phenomena and their contextual conditions. The business need for a CPQ solution results from an extensive and highly interdependent product or service portfolio, often offered by large companies or corporations. The typical situation before a CPQ implementation is characterized by a fragmented non-standardized process organization supported by a multitude of diverse non-integrated software tools leading to incalculable lead-times and high risk of error for quotes.

As we look at the CPQ process from a knowledge management perspective, we consider CPQ systems as a special type of knowledge management system (KMS). Creating a custom-tailored quote from generic offers based on a complex product and service portfolio requires the integration of expert knowledge from multiple business domains like product management, production and logistics, financial management, and sales. Hence, a CPQ implementation faces similar problems of knowledge acquisition and integration like the implementation of a general KMS. As we could not find literature on CPQ implementation, it seems appropriate to build on literature for implementing KMS. Because of the innovative character of CPQ systems, we selected the KMS

implementation model by Wang and Wang (2016) as a foundation for understanding the specific determinants of a successful CPQ implementation. Wang and Wang considered: 1) technology innovation; 2) organization; and 3) environment as the central influence dimensions for KMS implementation. We used this model as a framework for our case study design, guiding and directing the data collection as well as the analyzing process. The case data was collected during October and November 2019 from internal information resources like reports, documents and intranet web pages related to the examined CPQ implementation project. Additionally, we conducted three interviews with members of the implementation project team, who were selected due to their expert status. Interview partner #1 was the CPQ project manager, #2 the lead product modeler and #3 a head of deal management of one of the portfolio units. The interviews were conducted to explore and understand the implementation process as well as to collect experiences with the process. As a limitation, it should be noted that, although the CPQ system was already in productive use, the implementation project was not completely finished yet.

A Case Study on the Implementation of SAP CPQ at T-Systems

T-Systems International GmbH (TSI) with its headquarters in Frankfurt, Germany, offers a broad range of IT services for business customers grouped in four focus areas: Connectivity, Digitalization, Cloud & Infrastructure, and Security. TSI is currently located in over 20 countries, has approximately 37,500 employees and an external turnover of 6.9 billion Euros (2018). In the first quarter of 2018, the management board of TSI decided that the company's complex and time-consuming internal sales process should be streamlined and standardized to improve the company's competitiveness. It was also decided to implement a CPQ configurator for company-wide use. From the various CPQ vendors, SAP was selected who had acquired the vendor of the formerly independent CPQ product CallidusCloud shortly before. This product is now called SAP CPQ.

Initial Situation, Goals, and Challenges of the CPQ Implementation

Before the implementation of the CPQ system at TSI the configuration of an individual customer solution was done manually with only rudimentary support by various Excel sheets and other tools. There was no mechanism that automatically checked the configuration for plausibility and feasibility. The final quote documents were generated manually from document templates and text building blocks. The calculation of costs and prices represented a particularly complex and extensive undertaking. The prices were determined in the so-called Pricing Tool. The costs, however, were determined in another tool called ICT Calc. Both tools were based on Excel. Another tool then created an assignment between the cost blocks and the respective unit price. This assignment procedure was very extensive and complex. A change in one of the tables caused the need for manual adjustments in other tables. In addition, the individual tools were difficult to use and thus required specialists with expert knowledge for accurate and timely results. A larger and complex offering project required frequent iterations and consultations with several other experts. All quotations from the interviews are translated from German and redacted. In interview #3 it was pointed out:

“And that is exactly what takes a lot of time, resources and above all a high level of concentration and specialist knowledge.”

Facing this situation, TSI decided to overcome and eliminate the hurdles and sources of error in the quotation process by implementing a CPQ system across all portfolio units (PU) forming TSI's organizational structure. The solution should lead to a single standardized end-to-end process, automated and guided by the CPQ system. The respective product catalogs of the individual PUs were supposed to be stored in the CPQ and thus enable configuration and calculation across different PUs. In addition, the CPQ tool should completely replace the approximately 50 existing calculation tools and reduce the manual work through automated consolidation and standard templates. Furthermore, a seamless integration of the CPQ system with the ERP and CRM systems through standardized interfaces was pursued. An important goal was to reduce the time-to-offer as defined by the target that more than 90% of medium and minor deals should be handled in a maximum of five days.

Implementing CPQ with a Hybrid Agile Approach

The inner-organizational challenge in this project was to collect special product expertise, which is needed to equip the CPQ system with knowledge. Furthermore, acceptance of the new system should be created and propagated throughout the company. For TSI, the key to project success was an adaptive implementation process. A hybrid approach, meaning a combination of agile methods and classic phase models (Grey, 2011), was adopted for the TSI CPQ project. The classic waterfall model and the agile Scrum methodology (Schwaber & Sutherland, 2017) were used as basic ingredients for tailoring a TSI specific hybrid agile implementation process. At the beginning of the project, the topic of agility was still relatively unknown in the sales organization. However, the agile methodology is particularly suitable for projects in which no clearly formulated requirements and goals have yet been defined, which was the case with this project. The project management team had to recognize that the rest of the organization was not yet agile in its thinking and methodology but rather used to traditional concepts. The management, for example, asked for milestone planning, with clearly defined steps and responsibilities. The project management team was aware that a project of this complexity could hardly be managed successfully using traditional methods only. As the project manager stated in the interview:

"We created a hybrid approach: We defined milestones that determine when we want to achieve what in order to be able to serve traditional thinking in the group. But the way between the milestones, that is, the way we manage the project, is agile with Scrum."

The project manager stressed that this was an attempt to manage the balancing act between agile development and the still rigid structures within the sales organization. This way, the project team responded to the challenge of tearing down established paths and structures and breaking new ground.

The TSI project team structured the CPQ implementation in four phases (see Figure 1), each lasting several months and ending with explicit milestones. Each of these four phases was accomplished in four agile Sprints following the guidelines of the Scrum framework (Schwaber & Sutherland, 2017). At the end of each Sprint, a Sprint review was performed to assess and approve the progress made during the Sprint. Afterwards, based on the review, the next Sprint with further tasks to be implemented was planned. The SAP CPQ system was adapted to meet the requirements and processes of TSI in four successive development stages. All relevant stakeholders were involved

in the development process and their requirements were successively implemented in the CPQ system. From phase to phase, the developed functionalities were tested and stabilized with the support of selected test users.

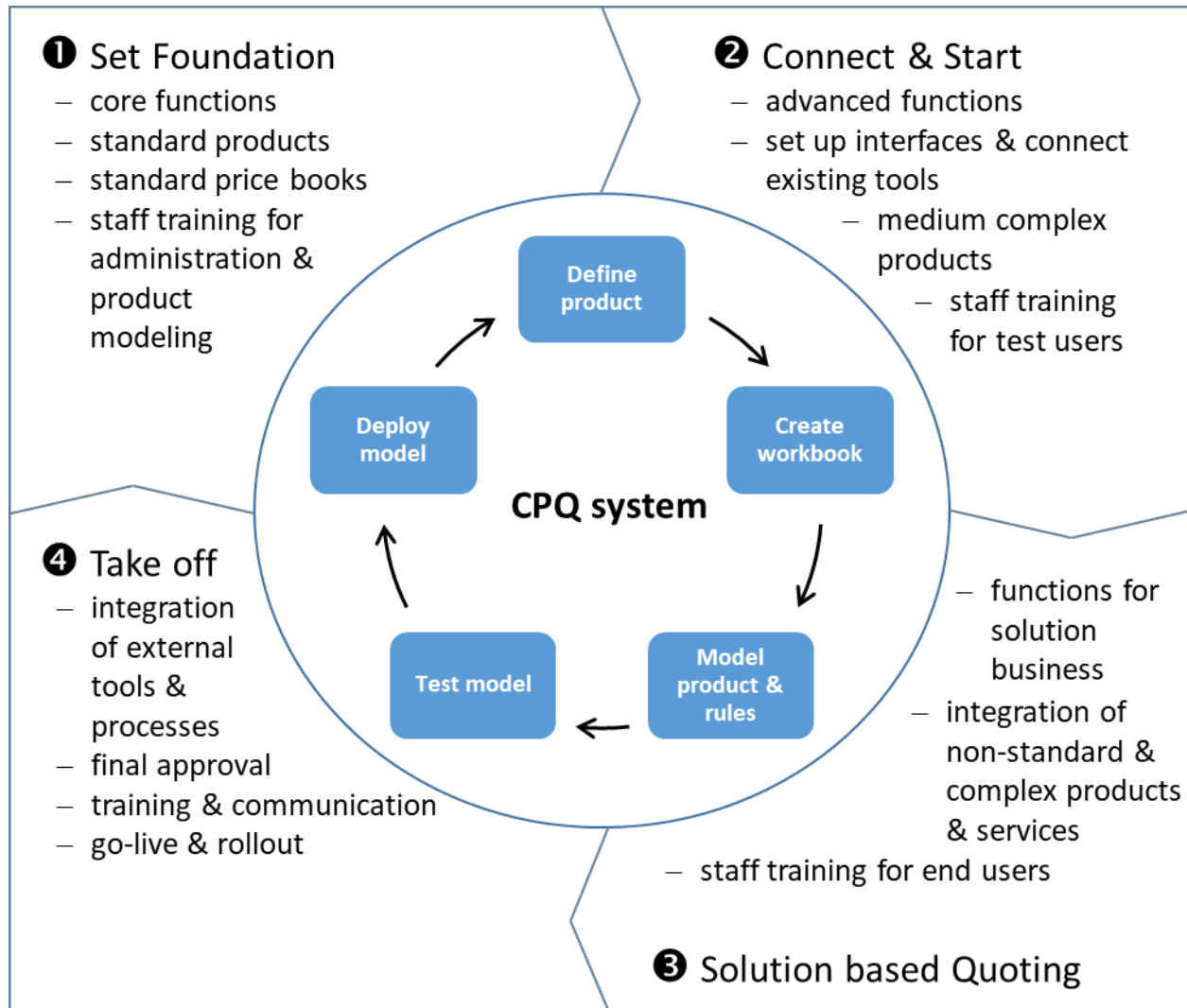


Figure 1. Process Model for the Hybrid Agile CPQ Implementation at T-Systems

These four phases of implementation of the CPQ software in the company were preceded by an additional pre-phase: the design phase. The company used this phase to agree on a suitable process model and to draw up the project plan in the form of milestones. The employees who were to take on project tasks were prepared for their activities in the project in this phase. Furthermore, the internal communication for introducing and explaining CPQ to the employees was planned and started. In order to strengthen the employee acceptance for the new CPQ solution its innovative character based on AI was emphasized in the internal explanation of the term CPQ:

But what does the term ‘CPQ’ mean exactly? The tool – whose acronym is a combination of the terms configure, price and quote – is based on research results from the field of Artificial Intelligence and fits into the existing system landscape. The CPQ platform provides product and solution models to the sales representative to assemble offers modularly. (TSI, 2019)

Knowledge Acquisition in the CPQ System

As shown in Figure 1, the focus of the implementation process was on creating a knowledge base on products in the CPQ system including their cost, price, and configuration rules. The workflow of knowledge acquisition and the according roles can be described as follows:

The main roles involved were: (1) the modeler who is responsible for mapping and testing the product in the CPQ system; (2) the offering manager responsible for the product; and (3) a developer responsible for software configuration, interfaces, and the necessary software changes. Before the modeler can start to map the product with its characteristics in the CPQ, s/he must know the properties and specifics of the product. The modeler receives the required information regarding costs and prices, as well as the properties of the product, from the offering manager. Based on the information received from the offering manager, the modeler creates a so-called workbook, that is, a document containing information and instructions for the model of the product in the CPQ system. After the workbook – including cost and price information as well as the configuration rules and conditions – has been completed, the respective product can be created in the development environment of the configurator component. During this step, product attributes are defined, the user interface is set up, rules and dependencies are defined, and cost and price tables are stored, as being the corresponding documents. At the beginning of the CPQ project, the completed workbook was passed on to an external modeler team of SAP for mapping it in the new CPQ system. However, after completing the modelling training internal TSI modelers have now taken on the modelling tasks.

The process continues with creating test protocols and testing the product model. The first test is performed on the developer environment. This test checks the rules, attributes, prices, and costs for completeness and configurability. Afterwards, the resulting quote is checked for correctness and whether the quote documents are generated correctly. The aim of the tests is, on the one hand, to check that the product is implemented in the CPQ system as it was specified or intended. On the other hand, the tests ensure that no errors occur during the configuration and that the entries in the quote and the attributes are valid. Furthermore, the protocol shows whether all the necessary information has been inserted via the mapping into the CPQ system. If unexpected errors occur during testing, they must be corrected. Otherwise, the product is ready to become transported from the developer environment to the User Acceptance Test (UAT) environment. This transfer to another environment is known as deployment. To allow the deployment to the UAT environment, a deployment sheet must be created. The deployment sheet documents which tables, documents, and rules belong to the product to be transported. This sheet is intended to prevent any data being lost during deployment and ensure that everything is transported to the new environment. After the deployment sheet has been created, the actual deployment takes place. All the data specified in the sheet is transferred and as a result, the product is available in the UAT environment.

After the UAT integration, the CPQ system is checked on plausibility, i.e. whether the deployment was successful, and all data has been transferred completely. In this case, the product is deployed from the UAT environment to productive operation. Initially, the product will only be made available to a small group of test users. It will be tested again, this time with the real prices. If all tests are passed, the product is made visible to all users and can be used for the offer and product configuration. Finally, the offer must be released in the CPQ system. For this purpose, the names and contact details of the releasing persons must be stored in the approval tables. An automatism within the configurator prevents official quote documents being generated without approval. In addition, the KPI table contained in the CPQ is updated and the new product configuration with its part and offer number is added. This table should then provide relevant key figures for the offers in productive operation and the reporting. All products that are mapped into the CPQ system go through the described procedure, which fills the system with knowledge that can be used for the guided preparation of an offer. As described in the workflow, the knowledge is supplied by knowledge holders and is essential for the correct representation of a product in the system.

Results and Findings

Although the CPQ implementation project of TSI ended in July 2019 with the successful go-live the implementation process is still ongoing. According to the agile principle of ‘inspect and adapt’, only a part of the whole product and service portfolio was transferred into the CPQ system before the company started to use it in daily business. The first experiences with using the system have reinforced the strategic goal of making the complete portfolio fully available via the CPQ system. Offering managers who used the new CPQ system reported “distinct” reductions of lead-time for the CPQ process (time-to-offer), depending on the complexity of the sales opportunity. The reported reductions range from 15-30 minutes to 2-3 weeks. These improvements became possible by reducing complexity through standardization. As the project manager pointed out:

“From, formerly, more than 600 single Standard Delivery Elements, with a huge amount of possible combinations, [we reduced] to approximately 30 predefined, scalable services, which are configurable by underlying rule sets in the CPQ system. In addition, the responsible deal manager no longer needs to get in touch with different resources, tools and processes in order to create prices via pricing tools, statements of work and offer documents. All that is now immediately available by ‘pressing a button’, when the configuration is finished.”

With respect to the concept of the general product lifecycle, the portfolio dynamics required a continuous process of adapting and maintaining the CPQ knowledge base. This was supported by the iterative design of the hybrid agile process model and emphasized by the project manager emphasized in the interview:

“And what cannot be overestimated is that we have managed to get the company to work and think in this agile way. Divisions have simply removed the complexity of their processes in order to be able to create offers faster. [...] We have accomplished much more than just creating a tool, we have changed people's mindset throughout this process.”

Furthermore, the hybrid agile process was able to increase transparency and strengthen mutual trust between the involved organizational units. As we learned from the project manager:

“It was also very important that we gave everyone the same voice, everyone should have the same right to be heard, even critics. That was probably a major success factor of the project. Mutual trust arose across the division boundaries. [...] Everyone was respected with his or her opinion and everything was decided by consensus. Sometimes you have to hold a discussion until you have a consensus. But in the end you have a decision that everyone supports.”

For reflecting the results of the TSI CPQ implementation, we once again draw on the KMS implementation model by Wang and Wang (2016). Along the three dimensions of this model, our case study showed:

- 1) The technology innovation of CPQ systems, especially the use of AI for configuration and recommendation, can stimulate strong expectations of perceived benefits. Especially for people being fond of IT, this effect can lead to a strong support of technology-based improvement initiatives, both on management and expert level. In this case study, this became relevant especially for ensuring the collaboration of expert knowledge holders.
- 2) The environment of the TSI CPQ project was positively influenced by the effect described under item 1. Furthermore, our case supports the findings of the study by Baltes et al. (2011) on the successful implementation of sales configurators. The authors state that the introduction of a preliminary phase prepares the employees better for the upcoming changes in the company and thus helps to achieve broad employee acceptance. In addition, a clear strategic positioning of the project from the beginning, as well as the provision of sufficient human and financial resources, were essential for the success of the CPQ implementation. TSI also carried out an initial phase before the actual implementation of the CPQ system and was thus able to prepare the organization for the upcoming changes and involve all the relevant knowledge holders. These preparation measures were considered key success factors in the interviews and should be taken into account when implementing projects in companies with similar structures and starting situations.
- 3) Regarding the organization, the case demonstrates how a hybrid agile implementation process can be applied to master the challenge of transforming and transferring both implicit and explicit expert knowledge from the minds of the knowledge holders into the codified knowledge base of a CPQ system. The iterative process consisting of time-boxed planning, executing and reviewing activities created transparency and trust that were the prerequisites for an open, respectful and creative collaboration of experts from multiple business domains. On the other hand, the classic waterfall phases with their milestones ensured the acceptance and support of traditional ‘command and control’ management that still can be found in many companies.

Because our case study only covered the period from start to end of the initial implementation project, it was not possible to collect more quantitative data on the achieved project success in relation to the formulated goals and objectives. Although the project success was positively evaluated by the interview partners, they could provide only a few concrete numbers on the achieved improvements of the CPQ project in terms of time and complexity.

Conclusions

We surveyed the implementation of a commercial off-the-shelf CPQ software from the process perspective along with technical and organizational challenges. Considering our initial research

question on the fitting implementation strategy and adequate knowledge-based system support, we can conclude that the multi-phase process model as depicted in Figure 1 and the hybrid agile approach of our showcase were supportive for the implementation success. To fully achieve the intended synergies and efficiencies in the quotation processes, the portfolio should be completely available in a CPQ system, which also requires a continuous process of adapting and maintaining the CPQ knowledge base.

CPQ-related technology innovations, such as AI and ML for configuration and recommendation, can stimulate the collaboration on management and expert level, in particular in the IT business. A preliminary project phase should involve all relevant knowledge holders and foster a broad acceptance. Furthermore, a clear strategic positioning of the project from the beginning is essential for the success of the CPQ implementation.

We also demonstrated how a hybrid agile process could master the challenge of transforming and transferring both implicit and explicit expert knowledge from the minds of the knowledge holders into the codified knowledge base of a CPQ system. The accompanying iterative process creates transparency and trust among experts from multiple business domains, while selected components of classic project management can involve the support from traditional management.

Since the presented process model was constructed based on a single case only, it needs further evaluation to demonstrate its appropriateness for general use. Major preconditions of the CPQ implementation at TSI included the company's industry (IT services and solutions), market (B2B), organization structure and size as well as the selected CPQ product (SAP). Hence, all three dimensions of Wang and Wang (2016) were addressed with a specific pre-implementation setup. Further research is necessary to explain and validate the causal relationship between the hybrid agile approach and the implementation success in more detail. A multiple-case study covering other industries and CPQ products could shed even more light on the mechanisms and success factors of a CPQ implementation.

References

- Abbasi, E. K., Hubaux, A., Acher, M., Boucher, Q., & Heymans, P. (2013). The anatomy of a sales configurator: An empirical study of 111 cases. In C. Salinesi, M.C. Norrie, Ó. Pastor (Eds), *Advanced Information Systems Engineering. CAiSE 2013. Lecture Notes in Computer Science*, 7908, 162–177. https://doi.org/10.1007/978-3-642-38709-8_11
- Asikainen, T., Soininen, T., & Männistö, T. (2002). *Representing software product architectures using a configuration ontology. Proceedings of the 4th International Configuration Workshop (associated with the 15th European Conference on Artificial Intelligence)*, 113–118.
- Baltes, G. H., Gard, J., & Mogck, A. (2011). Vertriebskonfiguratoren erfolgreich implementieren – Steigerung der Vertriebseffizienz durch den Einsatz von Produktkonfiguratoren. *Horizonte*, 37, 38–41.

- Bowen, J. (1986). Automated configuration using a functional reasoning approach. In A.G. Cohn, & J. R. Thomas (Eds.), *Artificial Intelligence and its Applications*, John Wiley & Sons.
- Bruno, J. (2017). The Forrester wave: Configure-price-quote solutions, Q1 2017. *Forrester Research*. <https://www.forrester.com/report/The+Forrester+Wave+ConfigurePriceQuote+Solutions+Q1+2017/-/E-RES132861#>
- Ferstl, O. K., & Sinz, E. J. (1996). Multi-layered development of business process models and distributed business application systems – An object-oriented approach. In König, W., Kurbel, K., Mertens, P., & Pressmar, D. (Eds.), *Distributed Information Systems in Business* (pp. 159–179). Springer. https://doi.org/10.1007/978-3-642-80216-4_10
- Franke, D. W. (1998). Configuration research and commercial solutions. *Artificial Intelligence for Engineering, Design Analysis and Manufacturing*, 12(4), 295–300. <https://doi.org/10.1017/s0890060498124022>
- Gill, J., & Mathur, G. (2019). Configure, price, and quote (CPQ) capabilities. *Deloitte Consult*. <https://www2.deloitte.com/content/dam/Deloitte/us/Documents/human-capital/us-consulting-cpq-capabilities.pdf>
- Grey, J. (2011). *The development of a hybrid agile project management methodology*. Doctoral thesis, Faculty of Computer Science, North-West University, South Africa.
- Hadzic, T., Subbarayan, S., Jensen, R., Andersen, H., Møller, J., & Hulgaard, H. (2004). Fast backtrack-free product configuration using a precompiled solution space representation. *Proceedings of the International Conference on Economic, Technical and Organisational Aspects of Product Configuration Systems*, 131–138.
- Heiskala, M., Tiihonen, J., & Soininen, T. (2005). A conceptual model for configurable services. *IJCAI Workshop on Configuration*, 19–24.
- Helo, P. (2006). Product configuration analysis with design structure matrix. *Industrial Management & Data Systems*, 106(7), 997–1011. <https://doi.org/10.1108/02635570610688896>
- Hotz, L., Felfernig, A., Günter, A., & Tiihonen, J. (2014): A short history of configuration technologies. In *Knowledge-based Configuration – From Research to Business Cases* (pp. 9–19). Morgan Kaufmann Publishers. <https://doi.org/10.1016/b978-0-12-415817-7.00002-5>
- Hvam, L., Pape, S., & Nielsen, M. K. (2006). Improving the quotation process with product configuration. *Computers in Industry*, 57(5), 607–621.
- Kling, J. (2019). Intelligence matters: AI powers SAP CPQ for optimal results. *SAP News Center*. <https://news.sap.com/2019/10/ai-powers-sap-cpq-optimal-results/>
- Klock, C., & Lewis, M. (2019). Magic quadrant for configure, price and quote application suites. *Gartner Research*. <https://www.gartner.com/en/documents/3970799/magic-quadrant-for-configure-price-and-quote-application>

- Kramer, B. M. (1991). Knowledge-based configuration of computer systems using hierarchical partial choice. *Proceedings of the 3rd International Conference on Tools for Artificial Intelligence*, 368–375. <https://doi.org/10.1109/tai.1991.167117>
- Krebs, T. (2006). Evolution of configuration models – A focus on correctness. *Proceedings of the ECAI Configuration Workshop*, 31–37.
- Ostrow, P. (2014). Configure / price / quote: Better, faster sales deals enabled. *Aberdeen Group*. https://cdn2.hubspot.net/hub/300410/file-2169344733-pdf/assets/CPQ_Endeavor_12.1_2014.pdf
- Schwaber, K., & Sutherland, J. (2017). *The scrum guide – the definitive guide to scrum: The rules of the game*. <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- Scott, T. (2018). CPQ software: Top 6 solutions and a case study. *TechnologyAdvice*. <https://technologyadvice.com/blog/sales/cpq-software-top-solutions/>
- Sorri, K., Kumpulainen, M., Seppänen, M., Dunne, M., & Huittinen, K. (2017). Prospects of CPQ: Evolving toward industry platforms. *Proceedings of the 9th International Workshop on Software Ecosystems*, 3–15.
- Tiihonen, J., Heiskala, M., Paloheimo, K.-S., & Anderson, A. (2006). Configuration of contract based services. *Proceedings of the ECAI Configuration Workshop*, 25–30
- TSI (2019). The official CPQ page: Configure.price.quote. *TSI Intranet Web Page*. <http://t-systems.telekom.de/cms/tsi-d/de/>
- Wang, Y.-M., & Wang, Y.-C. (2016). Determinants of firms' knowledge management system implementation: An empirical study. *Computers in Human Behavior*, 64, 829–842. <https://doi.org/10.1016/j.chb.2016.07.055>
- Yin, R. K. (2013). *Case study research – design and methods* (5th Ed.). Sage Publications.
- Zanker, M., & Tiihonen, J. (2008). Configuration and recommender systems: Two converging research fields. *IEEE Intelligent Informatics Bulletin*, 9(1), 3–4.

Authors' Biographies

Michelle Jordan graduated with a Bachelor of Science in business information systems from Leipzig University of Telecommunications (HfTL) as a cooperative student. During her studies, she was also employed at the cooperation partner T-Systems International GmbH. As part of her bachelor's thesis, Michelle surveyed the implementation of a CPQ configurator within T-Systems. Currently, she is working at the technical customer support department of Stay Informed GbR.



Gunnar Auth, Dr. is professor of information systems and e-government at Meissen University of Applied Sciences (HSF), Germany. He completed his diploma degree in business information systems at the University of Bamberg, Germany, and received a PhD degree in economics from the University of St Gallen (HSG), Switzerland. He started his professional career as an internal consultant at DaimlerChrysler where he later worked in several management positions in logistics, operations and quality management. Before assuming his current position, he was IT director and representative of the CIO board at Leipzig University. His research focuses on IT project management, IT service management and information management.



Oliver Jokisch, Dr.-Ing. is a professor of signal and system theory at the Leipzig University of Telecommunications (HfTL), Germany. He studied information technology at Technical University Dresden in Germany and Loughborough University in United Kingdom. Oliver graduated as a diploma engineer, and he holds a PhD degree in information technology from TU Dresden. His recent research is dedicated to artificial intelligence as well as to audio, speech and video communication. He co-founded the IT company voice INTER connect GmbH and the education and knowledge management firm IBWM GmbH in Leipzig.



Jens-Uwe Kühl is head of Solution Design Germany at T-Systems International GmbH. He studied mathematics at Humboldt University Berlin, Germany, and informatics at Technical University Dresden, Germany. The graduated computer scientist has been working in various technical, project and management positions at T-Systems with focus on managed data center and cloud services. At present, he is also responsible for the portfolio related implementation of T-Systems' central Configure Price Quote (CPQ) project with specific focus on Managed Infrastructure Services and Private Cloud.

